Stochastic model for Cognitive Radio Networks under jamming attacks and honeypot-based prevention

Suman Bhunia[1], Xing Su[2], Shamik Sengupta[3], Felisa Vázquez-Abad[4]
[1]sumanbhunia@gmail.com, [2]xsu@gc.cuny.edu,
[3]ssengupta@unr.edu, [4]FVazquez-Abad@gc.cuny.edu

[1,3]University of Nevada, Reno, NV, USA 89557
[2,4]City University of New York, NY, USA

International Conference on Distributed Computing and Networks
2014, Coimbatore, India

Introduction

- In Cognitive radio network, a secondary user (SU) goes through fixed periodic cycle of sensing and transmission period. We model this as FCFS queue with fixed server vacation. Sensing period is considered as vacation as the SU can't transmit packet in this period.
- Queue modeling with random server vacation has been studied extensively by the research community. However, fixed and periodic schedule of vacation leads added layer of complexity in the model.
- In this paper, we are going to present a queue model that deals with queue with fixed and periodic server vacation.
- Again, in the transmission period a secondary user might be assigned with other task such as honeypot. This leads to another kind of vacation. In this paper we also study honeypot scheduling in aspect of queue size of the node.

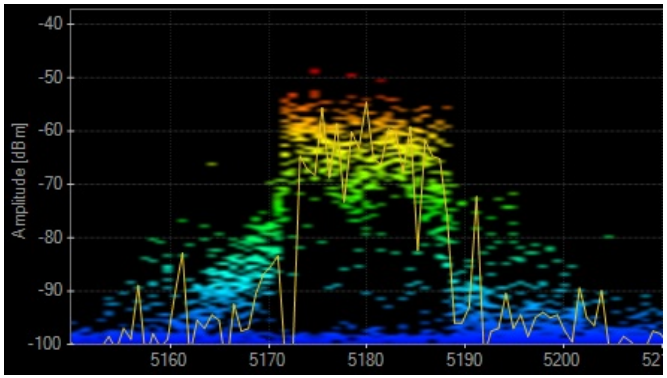
Cognitive Radio Network

- Cognitive Radio allows secondary user (SU) to use a spectrum when primary user (PU) is not using that spectrum.
- SUs sense for free channels in sensing period and can transmit on a channel on transmission period if the channel is free from PU.
- In a network, all SUs are synchronous for this cognitive cycle. Otherwise one SUs transmission will create false alarm to other SU.
- All SUs generate list of free channel in sensing period. Then central controller schedule which channel to be used by which SU in the transmission period.
- Each SU transmit packets in the transmission period over the channel dedicated by central controller.

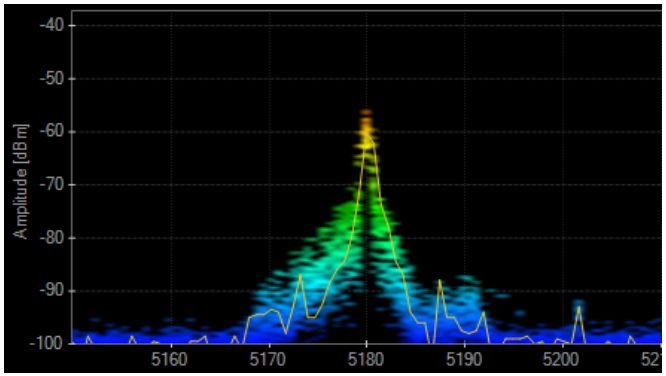
Jamming based DoS attack

- The nature of broadcasting electromagnetic signal on shared wireless medium is vulnerable to jamming based denial of service attack.
- While PUs are able to discourage attackers by means of heavy punishment, SUs are left vulnerable against malicious jamming / disruptive attacks
- Attacker emits jamming signal to create high interference to the communication of a legitimate wireless network and disrupt it.
- Advancement in software dened radio (SDR) and dynamic spectrum access (DSA) makes it even easier for smart malicious attacker to efciently find legitimate communication and disrupt.

Jamming attack example



PSD of normal data communication



PSD of narrow band jamming signal

- Two computers are configured to communicate over a WLAN (IEEE 802.11-a, channel 36, central frequency: 5.18 GHz)
- PSD shows high energy on 20 MHz channel and some leakage to neighboring channels.
- We begin transmitting a very narrow band jamming signal of 2MHz from a third machine, also on channel 36.
- In the presence of jamming signal, normal data communication is stopped. PSD only shows jamming signal energy.

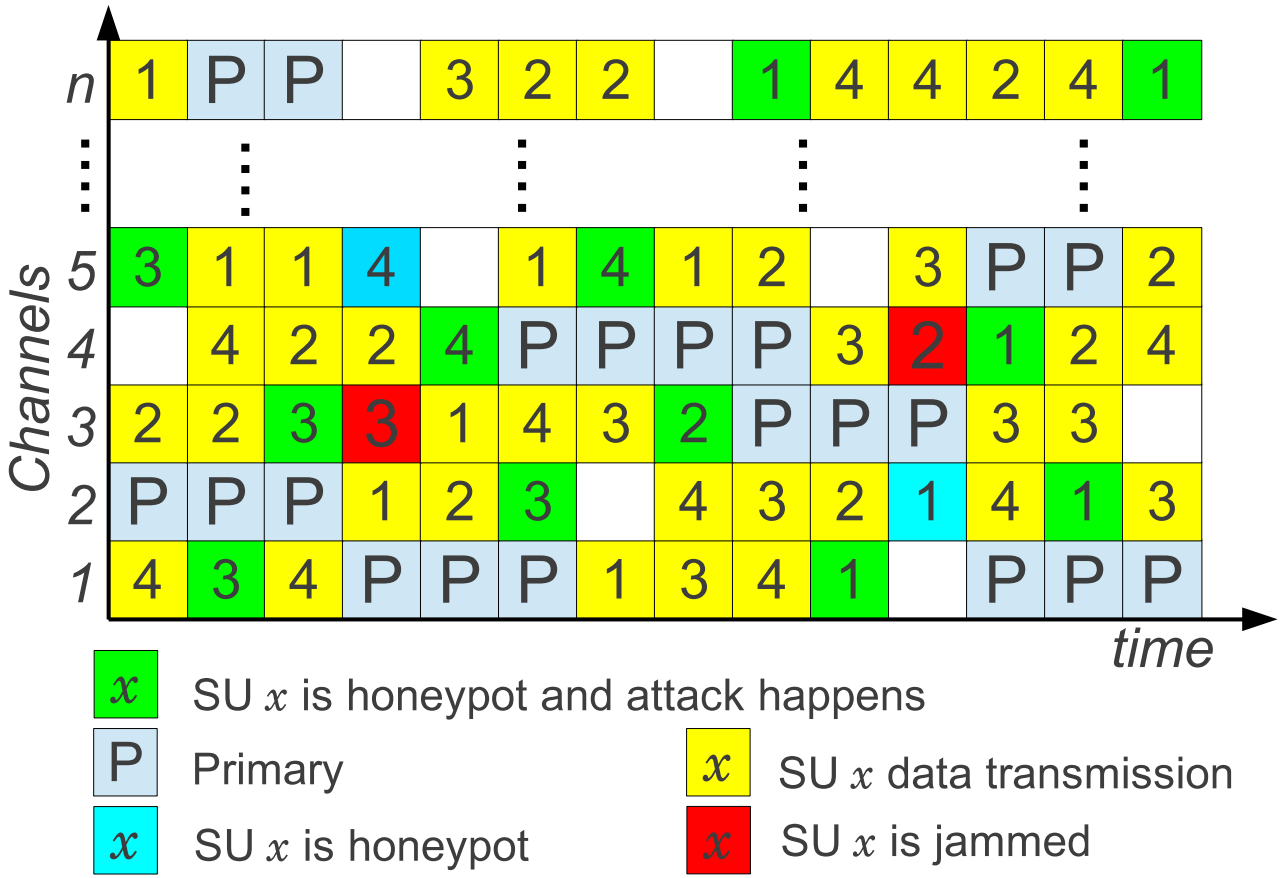
Use of Honeypot in avoiding attacks

- In Cybercrime defense, honeypots are being used as a camouflaging security tool with little or no actual production value to lure the attacker giving them a false sense of satisfaction thus bypassing (reducing) the attack impact and giving the defender a chance to retrieve valuable information about the attacker and attack activities.
- A smart attacker does not attack a channel blindly. It scans through all possible channels and choose one channel to attack that would give it best incentive out of jamming.
- It's strategy space includes but not limited to attacking channels with highest power, bandwidth, data-rate, particular modulation scheme etc.
- In CRN, honeypot mechanism can be deployed in one SUs which act as normal data transmission. In order to attract the attacker to this channel, the honeynode tries to obtain the attackers fingerprint from the past attack history.

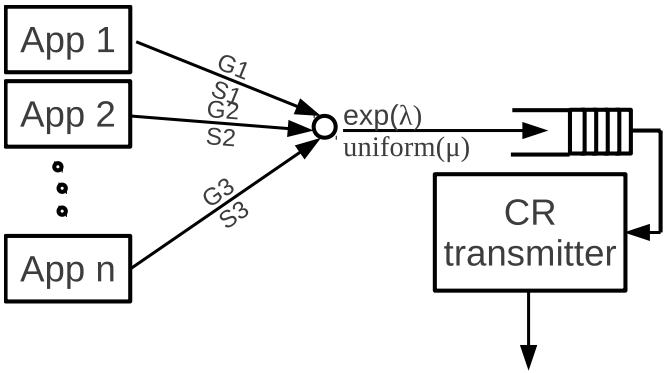
Use of Honeypot in avoiding attacks . . .

- The SU acting as honeypot node or honeynode transmit with transmission characteristics that is learned to be attacker's strategy. This lures the attacker to attack on honeynode's channel.
- Design of Honeypot is beyond the scope of this paper. We describe the design in another paper.
- In every transmission period, a SU is dedicated as honeynode. When acting as honeynode, that SU have to queue all its packet and transmit some garbage data.
- Sacrificing as honeynode causes delay to that particular SU. Choosing a honeynode from the network is a crucial problem to maintain the balance of system performance.
- Define a parameter, *Attractiveness of Honeypot* (ξ) as the probability of honeynode transmission is jammed given there is an attack.
- In this paper we present a theoretical model for calculating average queuing delay for a node. Then present honeynode selection strategy.

Channel Allocation by Central Controller



Mathematical Model: Queues



- Different application generates packets with different size and at random interval.
- These packets are queued upon generation.
- In transmission period if the node is not serving as honeypot, it transmits packets from queue.

- We model the queue of each SU as $M/G/1$ FCFS server with periodic "vacations". 'Service' means packet transmission. These vacations are: when SU is in sensing period, and when it is acting as honeynode.
- Poisson arrivals at each SU λ_i.
- Packet sizes $\sim U(\ell_1, \ell_2)$ in bytes, $\mathbb{E}(Y) = 1136$.
- Server needs $\delta = 10^{-7} msec$ to transmit each byte.
- Sensing period of 50 msec and transmission period of 950 msec.

Mathematical Model Contd ...

Assume that each queue is stable and call X the fraction of service that must be postponed at the start of a sensing period in steady state. Then:

$$\mathbb{E}(X) = \rho \left(\frac{\mathbb{E}(S) + \ell_1}{2} \right). \tag{1}$$

This result follows from straightforward calculation. Assuming that the queues are stable, the effective service rate for each of the SUs satisfies the equation:

$$\mu_i' = \mu \left(\frac{T_t - \rho_i \, 0.5(\mathbb{E}(S) + \ell_1)}{T_s + T_t} \right) (1 - p_i), \quad \rho_i = \frac{\lambda_i}{\mu_i'},$$

which yields an implicit equation for μ':

$$\mu_i'(T_t + T_s) = \mu \left(\mu T_t - 0.5 \frac{(\mathbb{E}(S) + \ell_1)\lambda_i}{\mu_i'} \right) (1 - p_i) \tag{2}$$

Stationary waiting time for infinite buffer model

Theorem

Suppose that i'th SU has incoming rate λ, and that it is chosen as a honeynode independently of the state of the queue, with long term frequency of p. Furthermore, assume that this queue is stable and ergodic and let X satisfy equations (1). Then the stationary delay in queue is:

$$W_q = \frac{R}{1 - \lambda \mathbb{E}(S)(1 + \Delta)}, \tag{3}$$

where the stationary residual service time is:

$$R = \frac{\lambda \, \mathbb{E}(S^2)}{2} + \frac{\mathbb{E}(T_s + X)^2 \, (1 - p) + \mathbb{E}(T_s + T_t + X)^2 \, p}{2(T_s + T_t)} \tag{4}$$

and the correction factor for the vacations is:

$$\Delta = \frac{T_s + \lambda \mathbb{E}(X) + p T_t}{(1 - p)(T_t - \mathbb{E}(X))}.$$

Proof: Idea

Poisson arrivals: system in stationary state.
$N_q \sim$ stationary distribution of the queue length.
$T =$ required service time for the customers in queue.

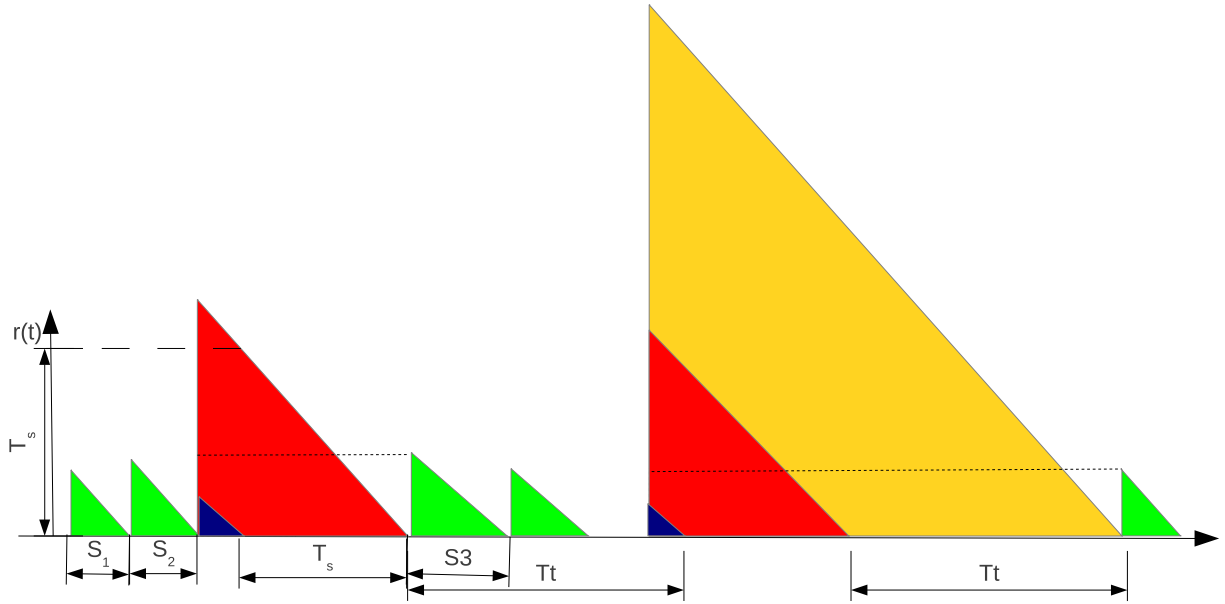
$$\mathbb{E}(T) = \mathbb{E}\left(\sum_{k=1}^{N_q} S_k\right) = \mathbb{E}(N_q \mathbb{E}(S)) = \lambda W_q \mathbb{E}(S).$$

Arriving customer has s stationary average waiting time:

$$W_q = \lambda W_q \mathbb{E}(S) + R + v_s(\, T_s + \mathbb{E}(\tilde{S})) + v_h\, T_t,$$

where v_s and v_h are the (expected) number of sensing and honeynode periods (respectively) that fall within the time required to transmit all the N_q customers in front of the new arrival.

Proof: residual time for next transmission



Here, green indicates packet transmission, red is Channel sensing, yellow is Serving as honeypot and blue indicates, the node postpone current packet transmission as it can not be done within transmission period.

$$R = \lim_{t\to\infty} \frac{1}{t} \int_0^t r(t)\, dt$$

$$= \lim_{t\to\infty} \frac{1}{t} \left[\sum_{i=1}^{M(t)} \frac{S_i^2}{2} + \sum_{i=1}^{V(t)} \left(\frac{\mathbb{E}(T_s + \tilde{S})^2)}{2} \right) + \sum_{i=1}^{H(t)} \frac{T_t^2}{2} \right]$$

Honeynode selection strategies

- Random selection: Centralized controller picks one SU randomly as honeypot, following a probability $p_i = \mathbb{P}(\text{node } i \text{ is chosen as honeypot}$ at each transmission period.
- Round robin: This is the simplest form of scheduling mechanism. Here honeypots are selected in circular fashion, selecting all SU's with equal frequency.
- State dependent: In this type of selection mechanism the centralized controller picks the SU according to the current values of the queues. These policies target a decrease in mean queueing delay.

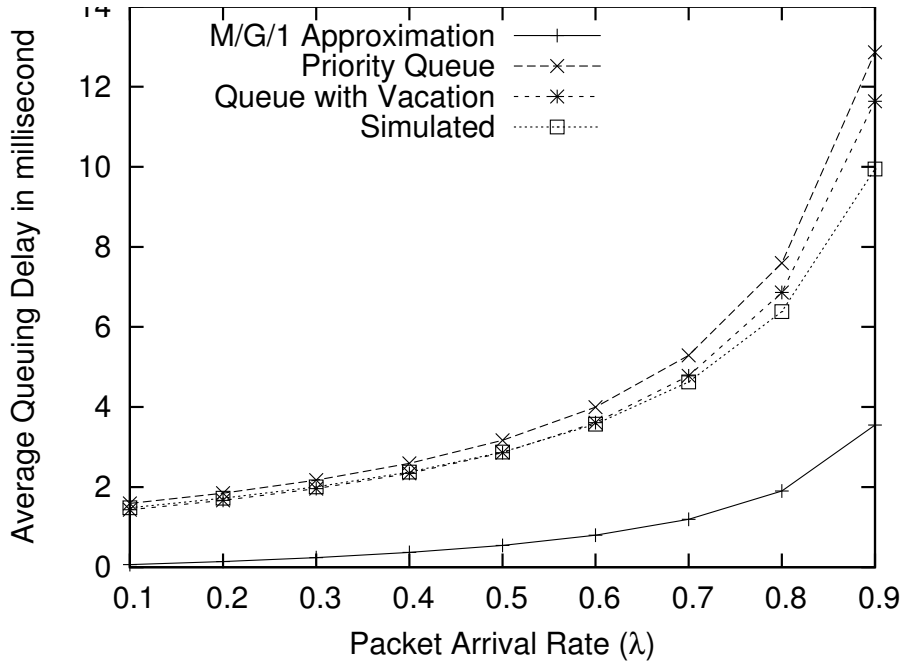
Simulator

We coded a discrete event simulation written in Python in order to compare the honeynode selection strategies. Default system parameters are given below:

Parameter	Symbol	Value
Number of SU	N	20
Packet Service Time	S_n	$\sim U(0.1, 1.7)$ msec
Sensing Period	T_s	50 ms
Transmission Period	T_t	950 ms
Number of attacks / slot		1
Number of honeynodes /slot		20
Number of replication		30
Simulated time		5000000 msec
Warm-up time		100000 msec

Comparison of approximations: without honeypot

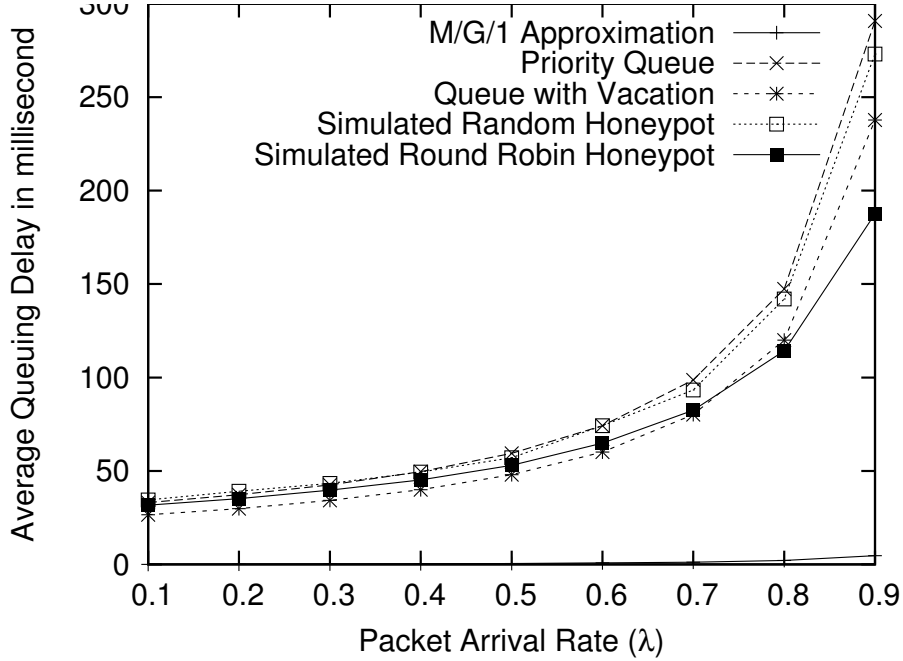
- M/G/1 Queue. A first crude approximation is to use the $M/G/1$ formulas with the effective rates λ and μ'. μ' can be calculated as in (2) $\mu' \in [1.05513, 1.05551]$.
- Priority model. A second approximation is based on a $M/G/1$ priority queue. The sensing operation is to be served in higher priority, and packet transmission is the lower priority job. This deals with services arriving according to Poisson process.



We can clearly see that our mathematical model gives closest approximation to the simulated result compared to already developed queue models.

Comparison of approximations: with honeypot

- One SU serves as honeynode.
- M/G/1 Queue. For random honeynode assignment ($p_i = 1/N$), the corresponding range is $\mu' \in [1.00283, 1.00320]$, ensuring stability for all queues.
- Priority model. The sensing operation is to be served in higher priority, serving as honeynode as second priority and packet transmission is the least priority job. This deals with services arriving according to Poisson process.



We can clearly see that our mathematical model gives closest approximation to the simulated result compared to already developed queue models.

Starvation probability

Strategies for honeypot allocation may include choosing the channel with largest probability of emptying. The particular case where all SU's have identical statistics (same arrival rates) is much simpler because it reduces to choosing the node with minimal queue size, and no further calculations are necessary. However in the more realistic scenarios when λ_i is not only different but perhaps even slowly changing, it may prove essential to be able to calculate specific trade-offs between choosing honeypots and avoiding attacks.

Probability that the queue empties within the current transmission period is $\mathbb{P}(\tau_u \leq T_t)$.
Classical risk theory : finite horizon "ruin probability".

Comparison of Strategies: infinite buffer

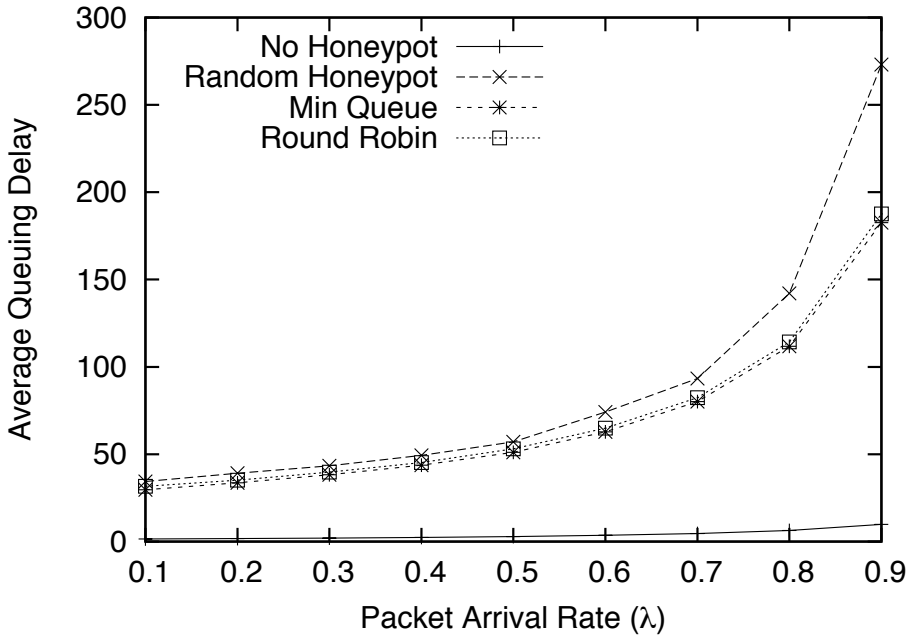


Figure: Average Queuing Delay in millisecond observed with infinite buffer and $\xi = 0.8$

- No packet drop for queue overflow, and therefore pdr is independent of λ.
- The only cause of packet drop is the jamming attack.
- Simulation results reflects that having no honeynode gives PDR of 0.05 and with one honeynode, PDR is 0.01 for all values of λ.
- Delay increases with λ and selecting SU with *minimum Queue* as honeynode gives better performance.

Comparison of Strategies: infinite buffer

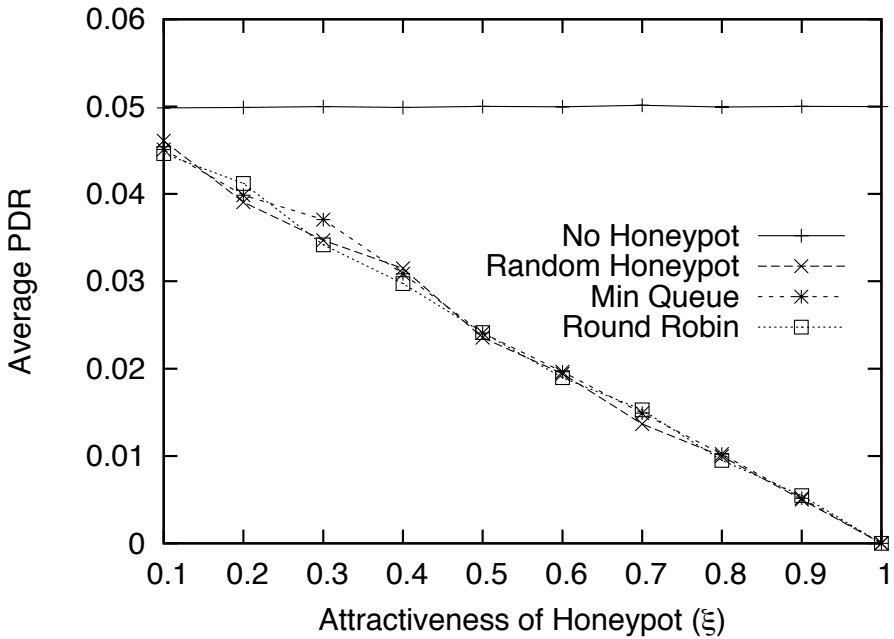
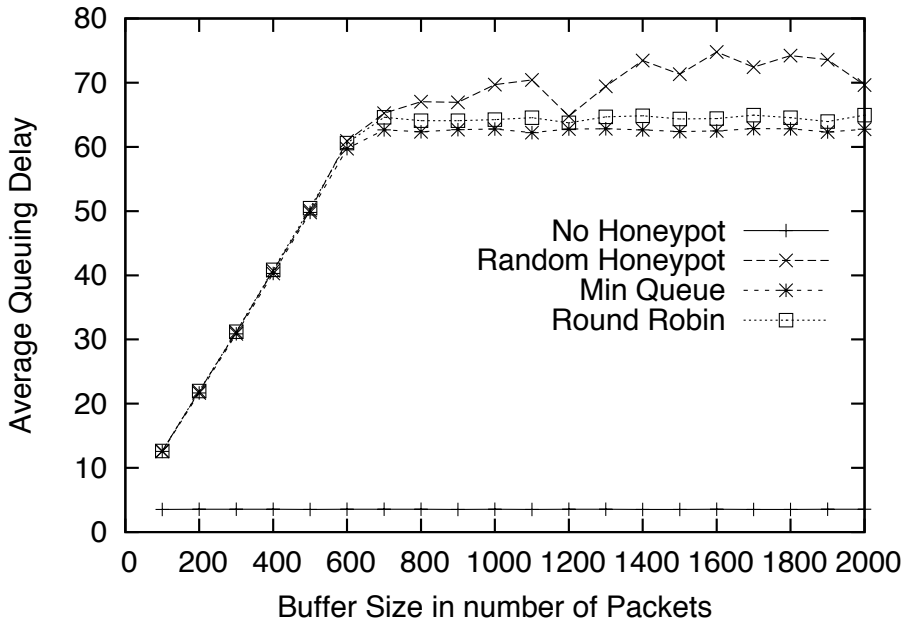


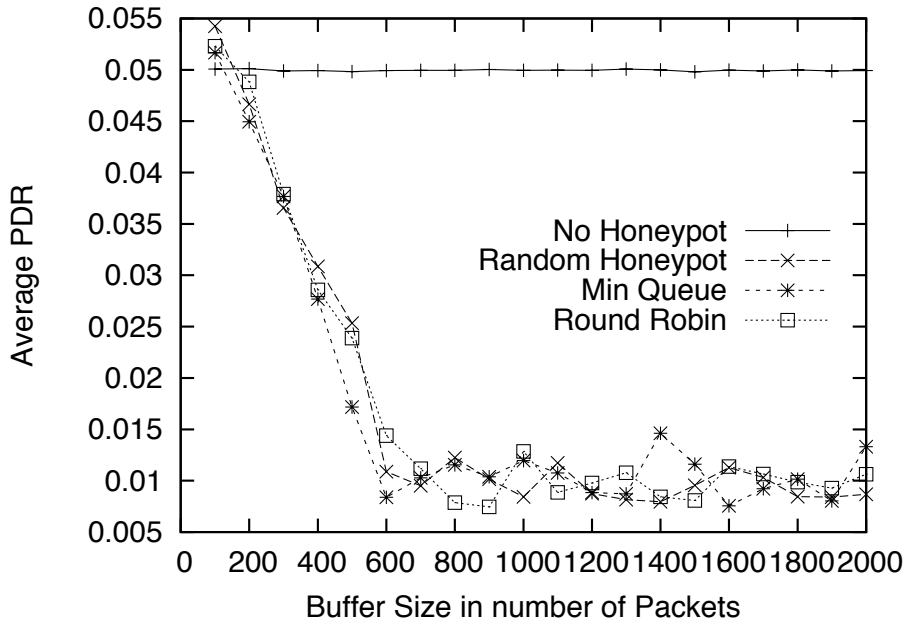
Figure: Average PDR of all SUs with infinite buffer and $\lambda = 0.6$

- Average queuing delay for no honeypot, random honeypot, minimum queue and round-robin selection schemes are 3.54 ms, 72.55 ms, 62.1ms and 64.62 ms respectively for all values of ξ.
- If θ_i = probability that i'th SU is attacked, then $PDR_i = \theta_i((1-p_i)+p_i(1-\xi_i))$. When $\theta_i = p_i = 1/N$ and $N = 20$ we obtain the linear function $0.05(1-0.05\xi)$.
- State dependent policy i.e. selecting honeynode based on minimum queue length ensures better performance.

Comparison of strategies: finite buffer



(a) Average Queuing Delay in msec

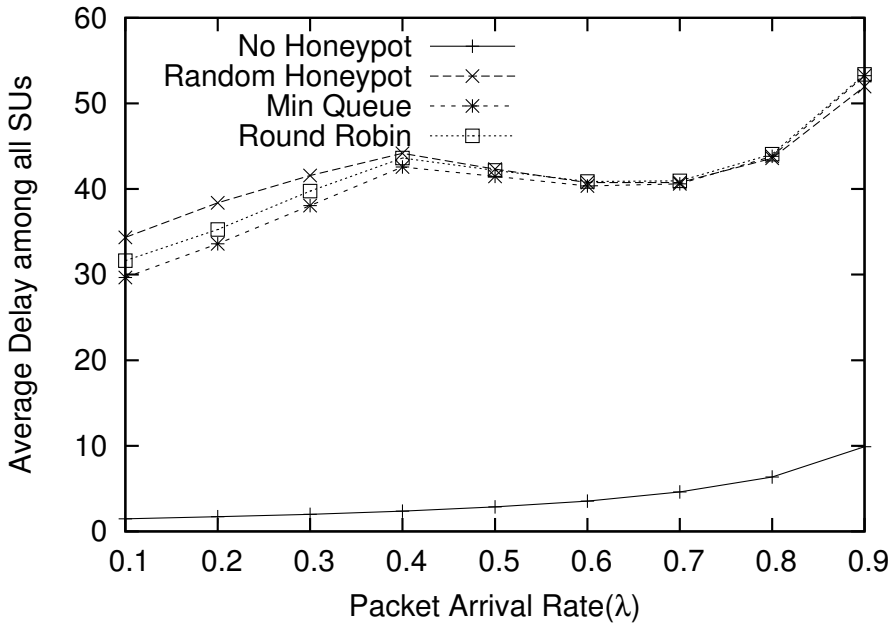


(b) Average PDR

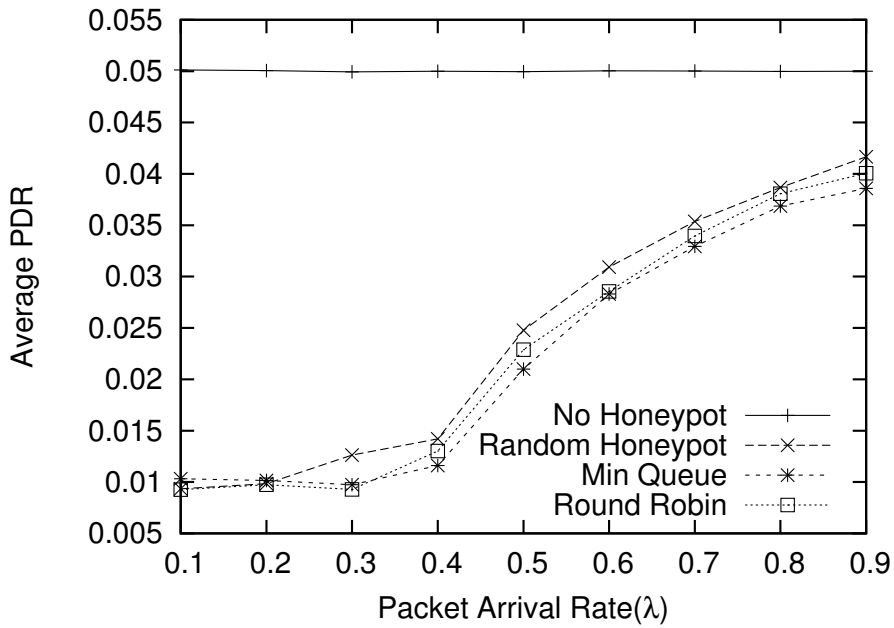
Figure: Results varying buffer size of SU with $\lambda = 0.6, \xi = 0.8$

On average $\lambda \times (2T_s + T_t) = 630$ packets would be queued when SU serves as honeynode. There would be queue overflow if buffer is smaller.

Comparison of Strategies: finite buffer



(a) Average Queuing Delay in millisecond



(b) Average PDR

Figure: Results for CRN varying λ with $\xi = 0.8$ and Buffer of 400 packets

When λ goes above the threshold $\mathit{Buffer\ Size}/(2T_S + T_t) = 0.381$, and the SU is serving as honeynode, SU accumulates many packets so that the queue overflows that causes significant amount of PDR.

Conclusion

- In this paper we presented a theoretical model to predict the performance of Cognitive Radio Network based on queuing model with fixed vacation.
- The model deals with the periodic sensing of cognitive cycle as fixed length vacation.
- Honeypot is well known to prevent jamming attack but assigning honeynode without considering queuing delay associated with it costs degradation of performance.
- Dynamic assignment of Honeynode is crucial from system performance perspective.
- We propose state dependent honeynode assignment scheme on every transmission cycle where the honeynode selection can be done by choosing the SU with lowest estimated queue size. This scheme performs well when all the SUs in network are having identical traffic load.

Future Work

- For Finite Buffer model, finding out lowest value of ξ below which, it is not worth using honeynode.
- Finding optimal balance between honeypot allocation and overall performance of network.
- Game theoretical framework for "intelligent" attackers?

Thank you for your attention.

Any Questions?